



Cross platform native mobile app development met C#



Marcel de Vries

Technology Manager, Info Support

marcelv@infosupport.com

@marcelv

#devNetNoord



Roy Cornelissen

IT Architect, Info Support

royc@infosupport.com

@roycornelissen



Willem Meints

Lead Developer, Info Support

willemm@infosupport.com

@wmeints



Agenda

- Mobile development intro
- Android platform (C#-Mono for Android)
- iOS platform (C#-MonoTouch)
- Windows Phone (C#-Silverlight)
- Multi platform architectuur
- Live demo implementatie register for session
- Lessons learned
- Q&A

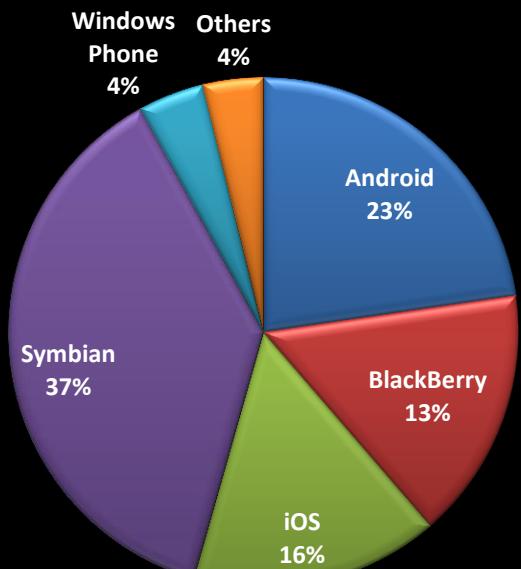
User Experience

- Emotie
- Gevoel
- Soms bijna een religie!
- Moet dicht bij de experience van het device liggen!

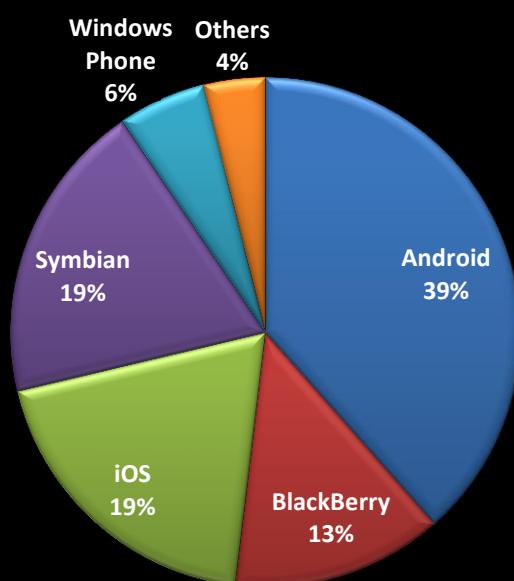


Market trends

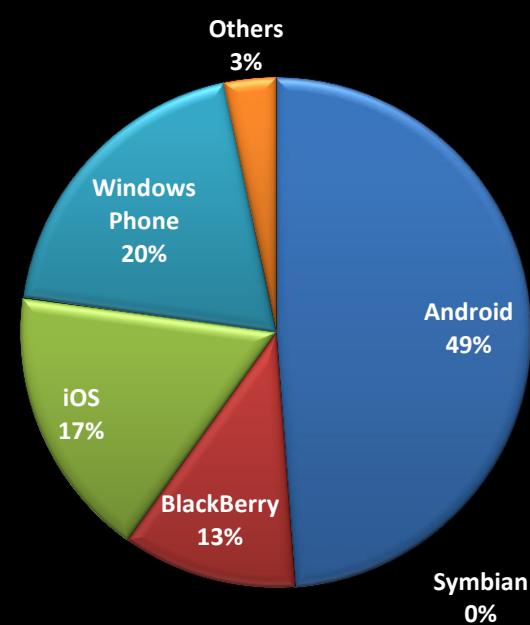
2010



2011



2015





Mogelijkheden om apps te bouwen



HTML 5 mobile websites



- Draait vanaf het web
- Maximum reach
- Geen native user interface
- Optioneel snelkoppeling op het device + Offline

HTML5 mobile apps



- Alleen verkrijgbaar via appstore
- Geïnstalleerd op het toestel
- Niet altijd een internet verbinding noodzakelijk
- Geen native user interface
- Meestal Magic box voor cross platform

Native mobile apps



- Alleen verkrijgbaar via appstore
- Geïnstalleerd op het toestel
- Niet altijd een internet verbinding noodzakelijk
- Native interface
- Maximum richness



Wat is native?





Wat is native?



Objective-C
C / C++

C / C++
Java
Different CPU
architectures



C# / Silverlight 3+



Native volgens ons...

- Native = gebruik van platform libraries & API's, ongeacht de taal
- Native = User experience 100% in lijn met native device experience



C# en .NET als bindende factor
3 platforms, 2 frameworks, 1 taal
Write once, run many?

Bestaande skills benutten!



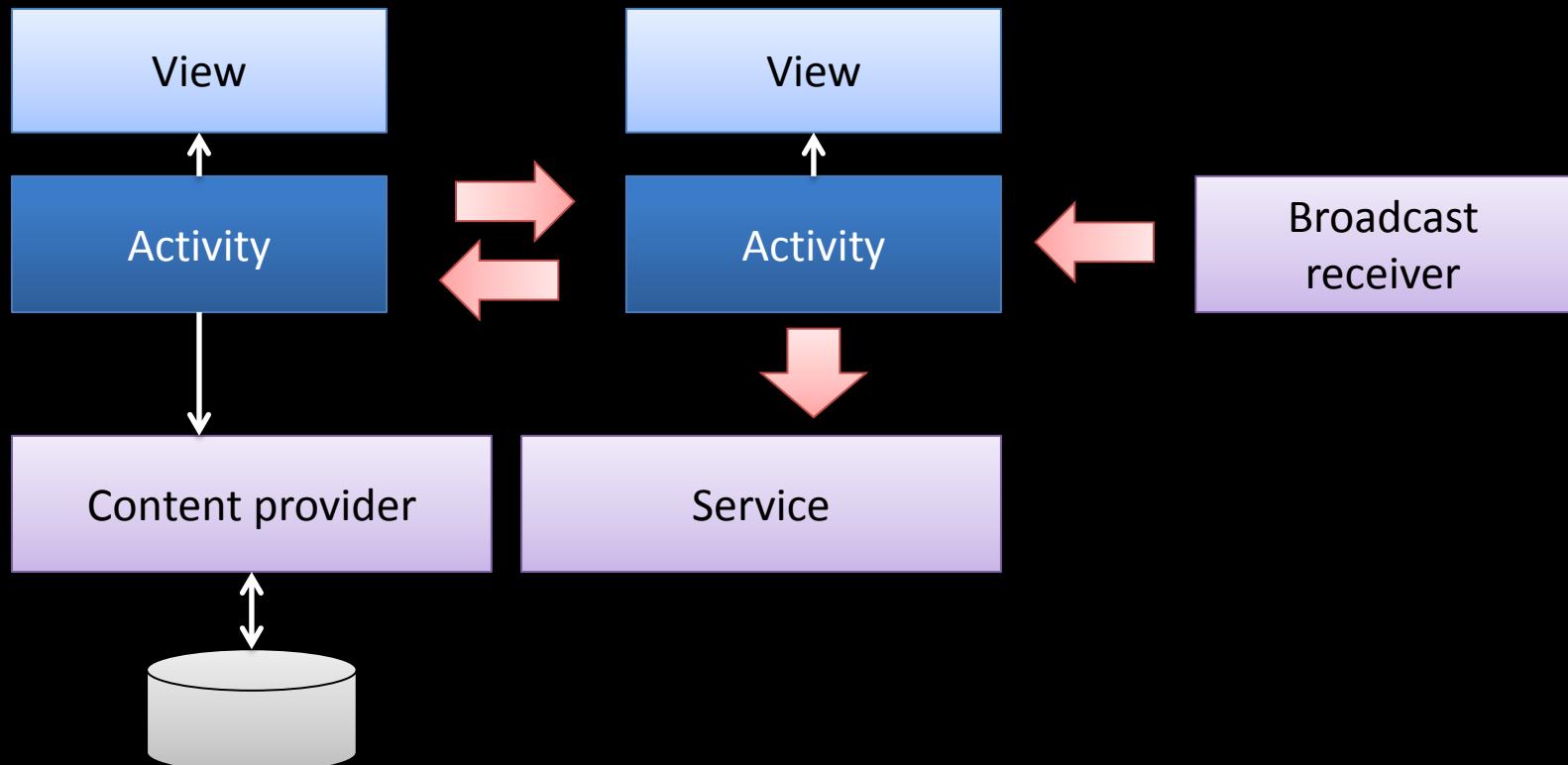


Mono for Android

- .NET implementatie voor Android
 - Mono 2.10
 - Support een aantal Silverlight API's (Isolated storage)
 - OpenGL support
- Bindings tegen Android
 - Activities
 - Services
 - ContentProviders
 - WebKit
- Versie 4.0.4 net uit!
 - Windows: Visual Studio 2010
 - Mac: MonoDevelop
- Primair UI model: "MVC"

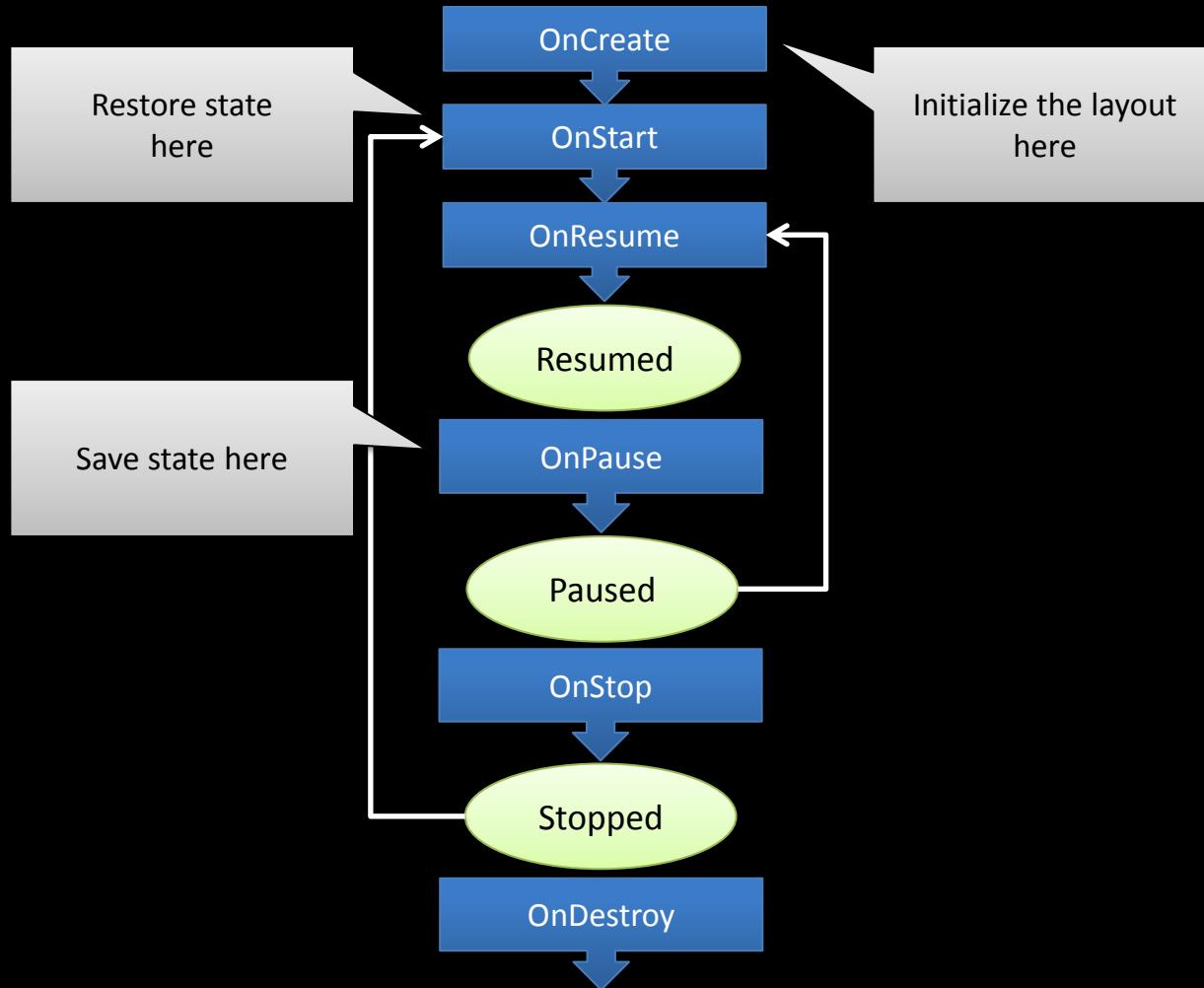


App model





Activity lifecycle





Mono for Android

It's cute, can we see it?



- Voor mobile geoptimaliseerde versie Mac OSX
 - Cocoa Touch framework (MVC architectuur)
 - Apple developer tools
 - XCode + Interface Builder
 - Objective-C of C/C++
 - Enterprise features
 - “Section 3.3.1: Applications must use documented APIs”A portrait photograph of Steve Jobs, co-founder of Apple, looking slightly to the right. He has short, light-colored hair and is wearing a dark shirt.

“Okay, how shall we do this? Use any tool you want, just as long as you ONLY USE the resulting apps APPLE DEV. NO VODKA, NO ANY CODE TOOLS!”

3.3.1 Applications may only use Documented APIs in the manner specified by Apple. You must not use or call any private APIs. Applications must be originally written in Objective-C, C, C++, JavaScript as executed by the iPhone OS WebKit engine, and any code written must be valid Objective-C. You may compile and directly link against the Documented APIs (e.g., Apple’s Foundation API) to Documented APIs through an intermediary translation or compatibility layer or tool are prohibited).



“Okay, okay, use any tool you want, just as long as the resulting app/app/Phone/Dev/Moad app code looks like this:

3.3.1 Applications may only use Documented APIs in the manner specified by Apple. You must not use or call any private APIs. Applications must be originally written in Objective-C, C, C++ or JavaScript as executed by the iPhone OS WebKit engine, and any code written in C, C++, and Objective-C may only be linked against the Documented APIs (e.g., ADT or any other link to Documented APIs through an intermediary translation or compatibility layer or tool are prohibited).

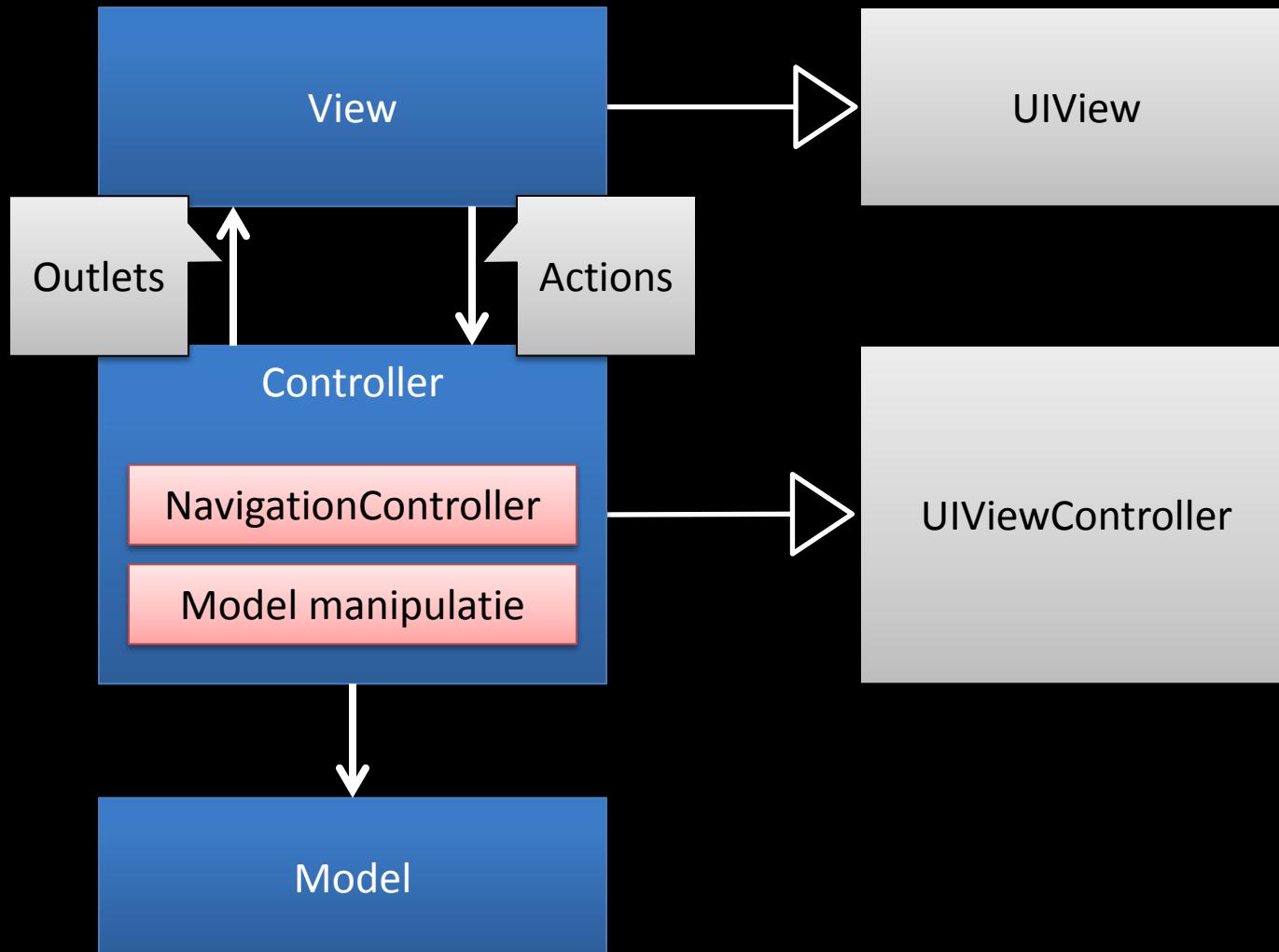


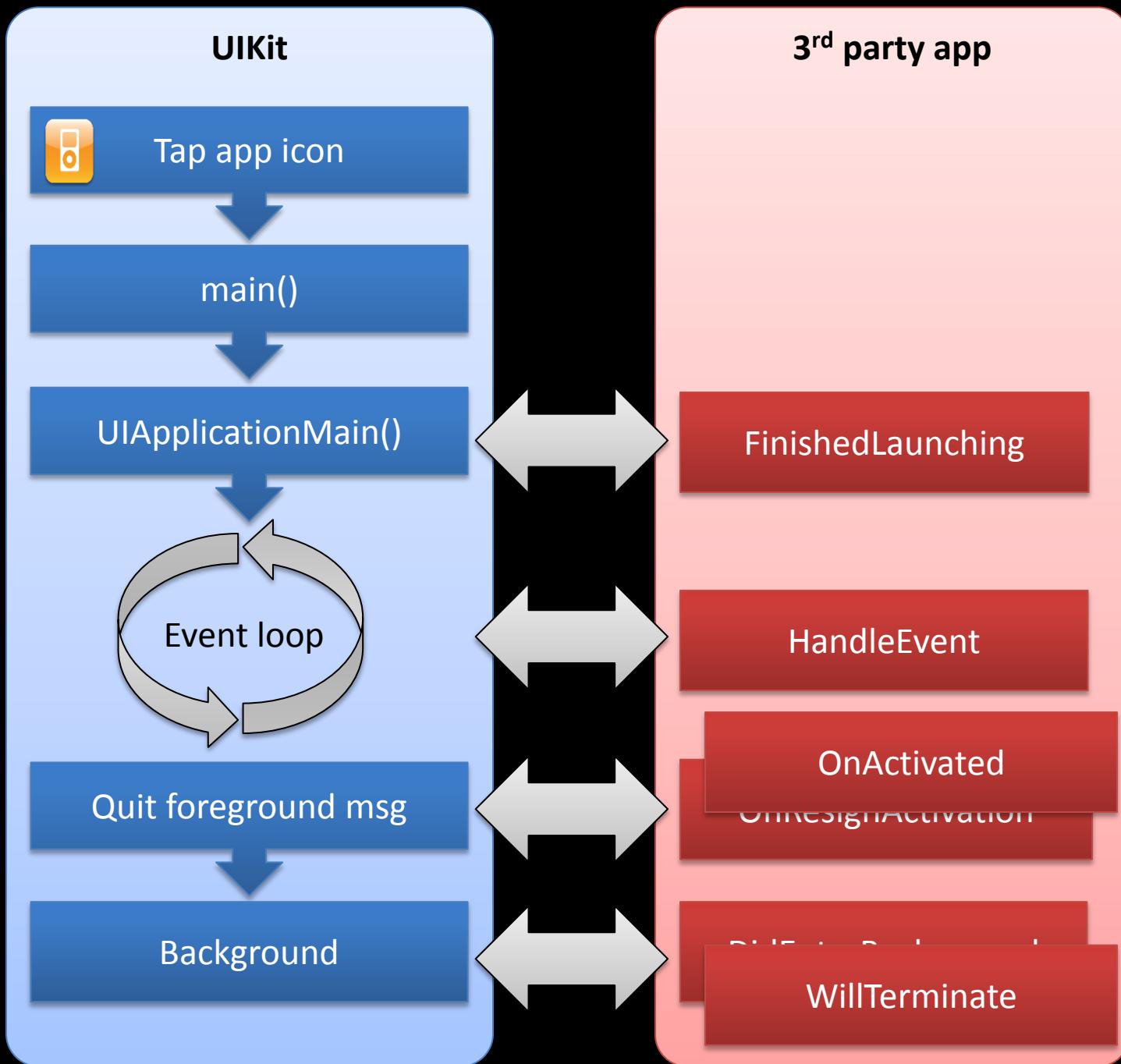
MonoTouch

- .NET implementatie voor iPhone en iPad
- Standaard .NET features
 - C# 4.0
 - Garbage collection
 - LINQ
 - System.Data
 - Parallel FX
- Bindings met native API
 - UIKit
 - CoreGraphics
 - CoreAnimation
 - AddressBook
 - CoreText
 - AVFoundation
- Primair UI model: MVC
- MonoTouch op elk iDevice in elke Apple store!



App model







MonoTouch

Show me the money!





Windows Phone 7

- Mobiel besturingssysteem van Microsoft
- Apps bouwen met Silverlight of XNA
- Featureset is Silverlight 3.5+
- Primair UI model: MVVM
- Tango update: ~mei 2012



Hardware Specification

Consistente set hardware specs door Microsoft



Capacitive touch

4 or more contact points

Sensors

A-GPS, Accelerometer, Compass, Light, Proximity

Camera

5 mega pixels or more

Multimedia

Common detailed specs, Codec acceleration

Memory

256MB RAM or more, 8GB Flash or more

GPU

DirectX 9 acceleration

CPU

ARMv7 Cortex/Scorpion or better

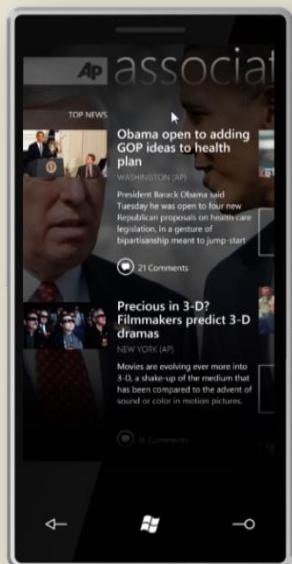
..... Hardware buttons | Back, Start, Search

Two Flavors of Applications



Microsoft®
Silverlight™

- Modern XAML/event-driven application UI framework
- Rapid creation of visually stunning apps
- Metro-themed UI controls
- 500,000 developers spanning Windows and web

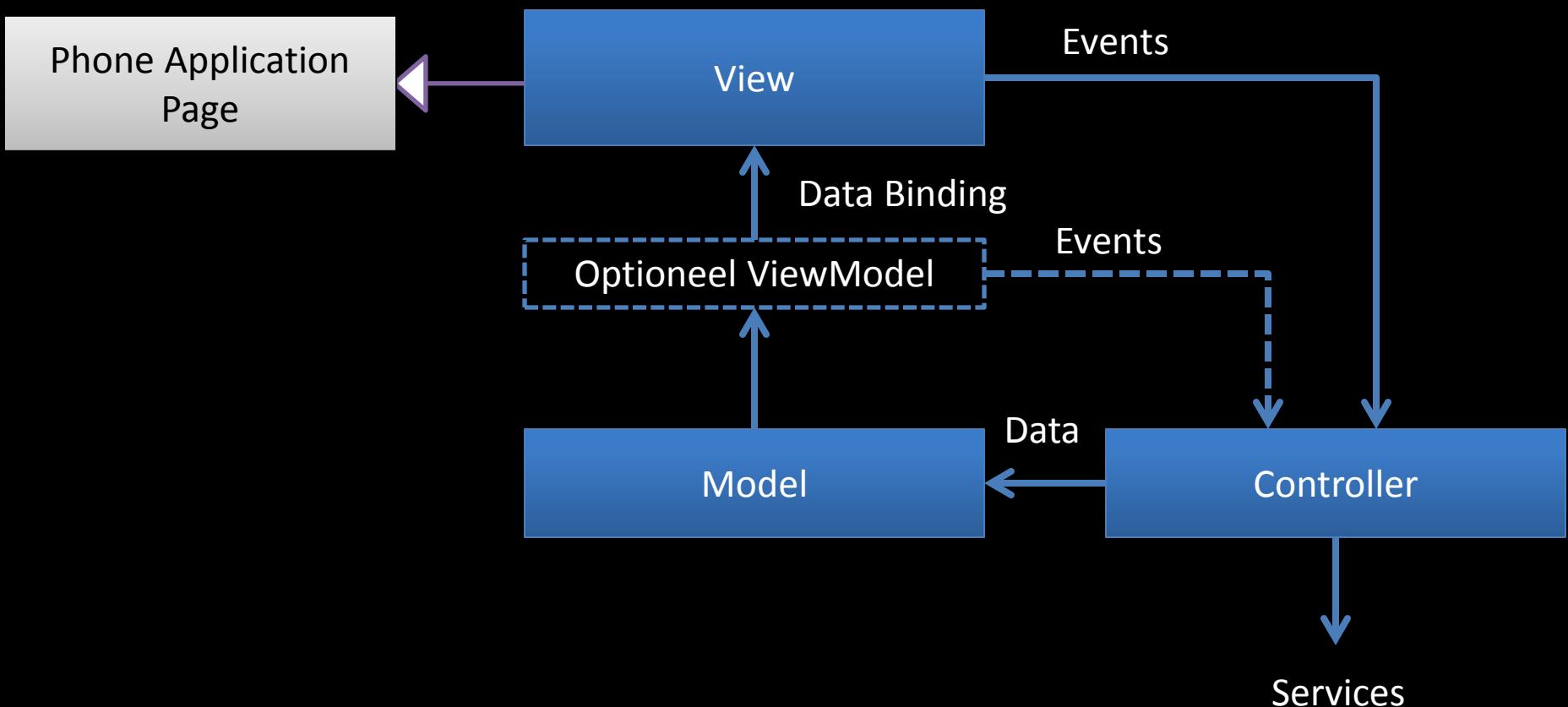


- High performance game framework
- Rapid creation of multi-screen 2D and 3D games
- Rich content pipeline
- Mature, robust, widely adopted technology spanning Xbox 360, Windows, and Zune

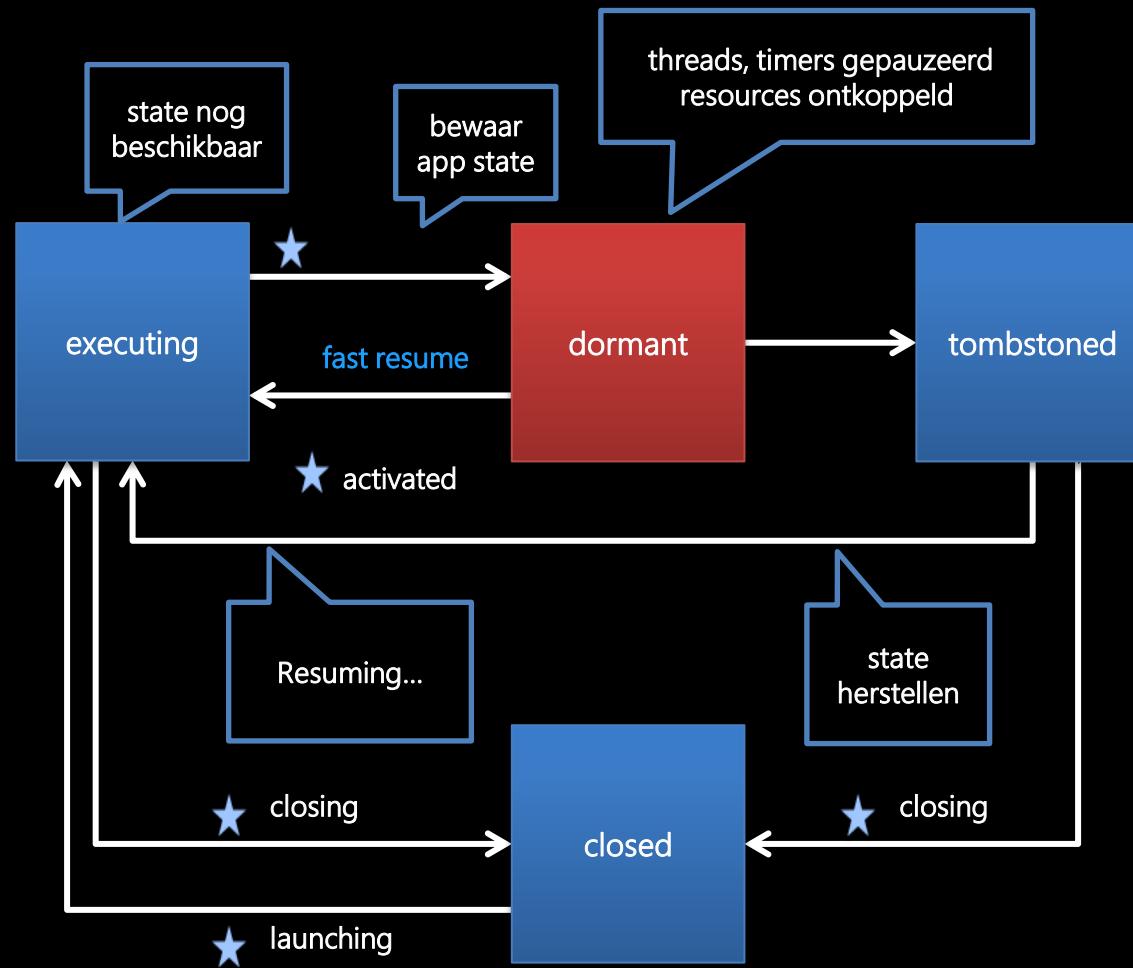




WP7 app model



WP7 app lifecycle



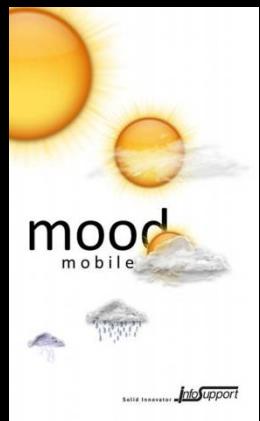


Windows Phone

QUIT TALKING, DEMO!



Mood App





Slaap Coach





www.LastSeats.com





Events app

DEMO

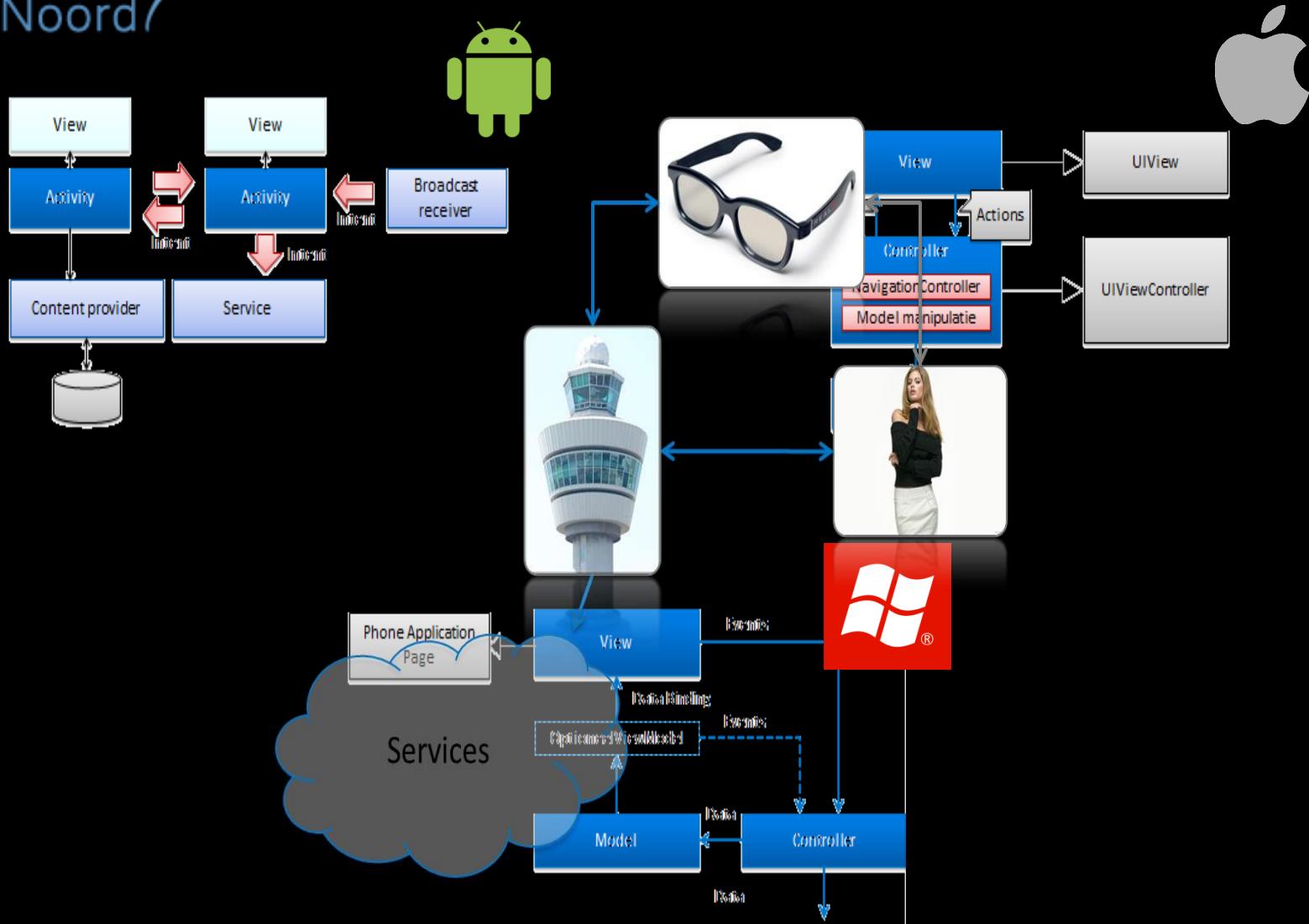




DESIGN PATTERNS VOOR HERGEBRUIK

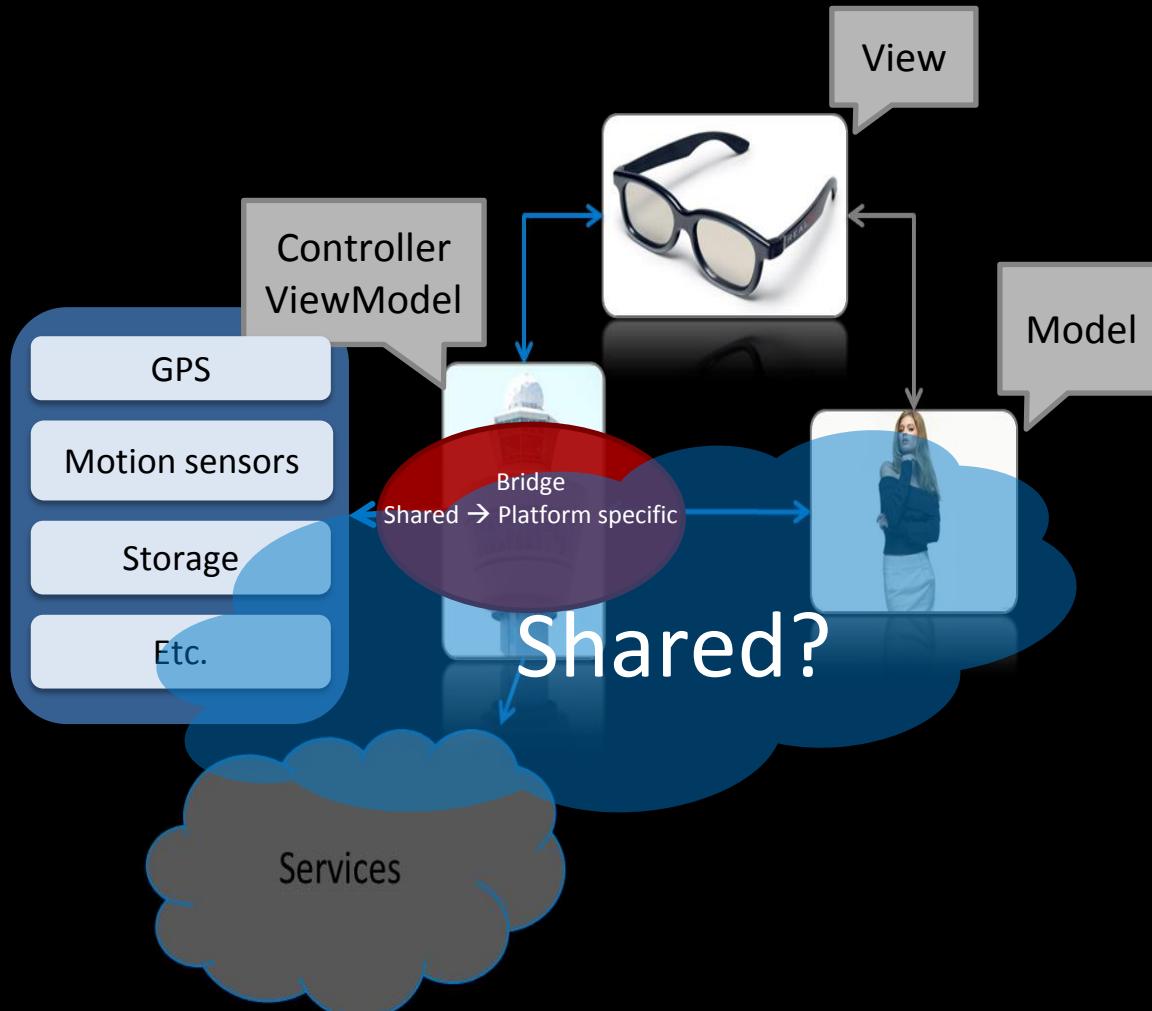


Define architecture for reuse



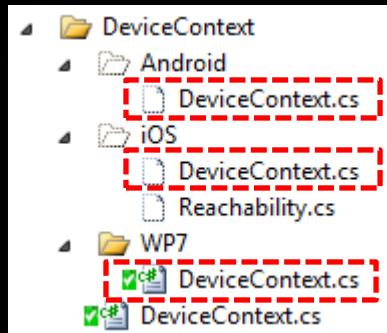


Define architecture for reuse





Device abstraction



```
/// <summary>
/// Executes an action on a foreground thread
/// </summary>
/// <param name="action">Action to execute</param>
partial void InternalRunOnForegroundThread(Action action)
{
    Deployment.Current.Dispatcher.BeginInvoke(action);
}
```



```
/// <summary>
/// Executes an action on a foreground thread
/// </summary>
/// <param name="action">Action to execute</param>
public void RunOnForegroundThread(Action action)
{
    InternalRunOnForegroundThread(action);
}

/// <summary>
/// Executes an action on a foreground thread
/// </summary>
/// <param name="action">Action to execute</param>
partial void InternalRunOnForegroundThread(Action act
```

```
partial void InternalRunOnForegroundThread(Action
{
    _rootActivity.RunOnUiThread(action);
}
```

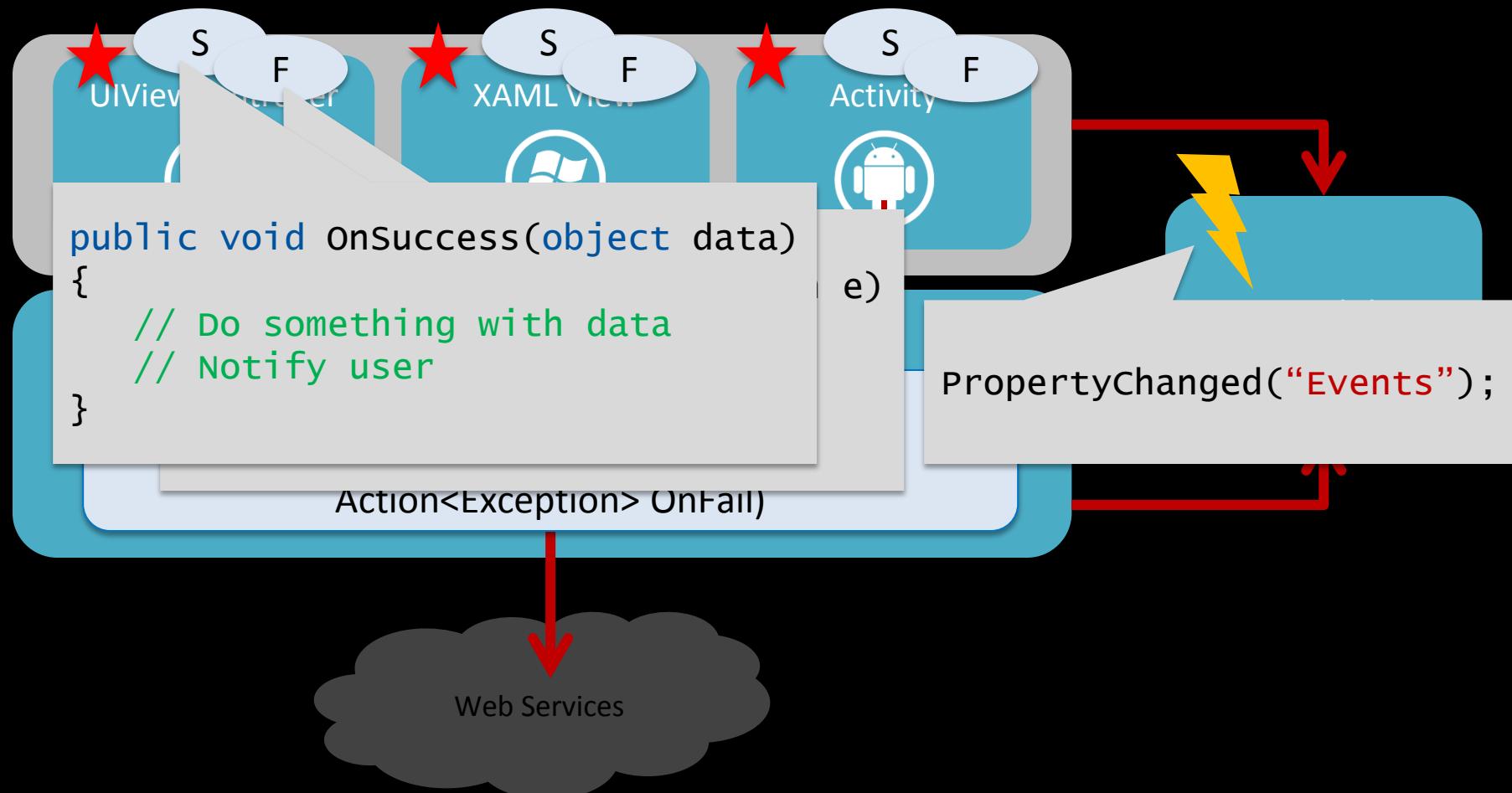


```
/// <summary>
/// Executes an action on a foreground thread
/// </summary>
/// <param name="action">Action to execute</param>
partial void InternalRunOnForegroundThread(Action action)
{
    UIApplication.SharedApplication.InvokeOnMainThread(() => action());
}
```





Brug tussen UI en shared logic



A clear glass bowl filled with red jello cubes, arranged in a somewhat circular pattern. The bowl is placed on a dark wooden surface. In the background, there are some out-of-focus elements, possibly a window and some shelves.

**THE PROOF OF THE PUDDING
IS IN THE EATING**



Lessons learned

- Android == Wild Wild West of mobile
- Test ALWAYS on MULTIPLE devices!
- Mono Release cycle only small lag with vendor tools
- UX design essential for apps
- Good design up front of the app essential for code sharing
 - First attempt 10% code share ☺
 - Now we achieve 70-80% code sharing!





Bedankt!



Marcel de Vries

marcelv@infosupport.com

@marcelv



Roy Cornelissen

royc@infosupport.com

@roycornelissen



Willem Meints

willemm@infosupport.com

@wmeints